



**UNIVERSIDAD TECNOLÓGICA
NACIONAL
FACULTAD REGIONAL
RESISTENCIA**

INGENIERÍA EN SISTEMAS DE INFORMACIÓN

MATERIA: SISTEMAS DE GESTIÓN DE BASES DE DATOS

NIVEL: 3

CICLO LECTIVO: 2024 – 2DO. CUATRIMESTRE

DOCENTES:

PROFESOR: I.S.I. ANDRÉS PABLO FANTÍN

J.T.P.: I.S.I. JUAN CARLOS FERNÁNDEZ

AUXILIAR ADSCRIPTA: LUCIANA CAMPESTRINI

Grupo n.º2:

- Arduña Zago, Agustín Juan Luis
- Berón de Astrada, Santiago Agustín
- Fernández, Facundo Nahuel
- Kinweiler, Víctor Fabricio
- Schefer, Mauricio Nicolás
- Velazco Gez Schegtel, Juan Ignacio

1) Escribir un documento XML para registrar préstamos de una biblioteca.

a. En el documento se indicará:

1. El nombre y apellidos del bibliotecario responsable del préstamo.
2. Fecha del préstamo y de devolución.
3. Datos del lector (id, nombre, apellidos, teléfono y dirección).
4. La dirección se dividirá en tipo de calle (que puede ser calle o avenida), nombre calle, número, piso y letra, código postal, localidad y provincia.
5. Un máximo de tres ejemplares en préstamo. Para cada uno de ellos: el número de registro, título, autor(es).
6. El préstamo tendrá un atributo numérico que servirá como identificador.

```
<?xml version="1.0" encoding="UTF-8"?>
<prestamos>
  <prestamo id="1">
    <bibliotecario>
      <nombre>Juan</nombre>
      <apellidos>Pérez García</apellidos>
    </bibliotecario>
    <fecha>
      <prestamo>2024-10-26</prestamo>
      <devolucion>2024-11-10</devolucion>
    </fecha>
    <lector>
      <id>L12345</id>
      <nombre>Paul</nombre>
      <apellidos>Newman</apellidos>
      <telefono>123456789</telefono>
      <direccion>
        <tipo>Avenida</tipo>
        <nombre>Figuerola Alcorta</nombre>
        <numero>7597</numero>
        <piso>2</piso>
        <letra>A</letra>
        <codigoPostal>28013</codigoPostal>
        <localidad>Buenos Aires</localidad>
        <provincia>Buenos Aires</provincia>
      </direccion>
    </lector>
    <ejemplares>
      <ejemplar>
        <numeroRegistro>1001</numeroRegistro>
        <título>El Quijote</título>
        <autores>Miguel de Cervantes</autores>
      </ejemplar>
    </ejemplares>
  </prestamo>
</prestamos>
```

```

    </ejemplar>
    <ejemplar>
      <numeroRegistro>1025</numeroRegistro>
      <titulo>El amor en los tiempos del cólera</titulo>
      <autores>Gabriel García Márquez</autores>
    </ejemplar>
    <ejemplar>
      <numeroRegistro>2030</numeroRegistro>
      <titulo>1984</titulo>
      <autores>George Orwell</autores>
    </ejemplar>
  </ejemplares>
</prestamo>
</prestamos>

```

b. Verificar que esté bien formado.

El documento está bien formado porque cumple con las 10 reglas y pautas de XML con respecto a la sintaxis de la misma. Con lo comprobamos en el website www.xmlvalidation.com

Please copy your XML document in here:

```

<?xml version="1.0" encoding="UTF-8"?>
<prestamos>
  <prestamo id="1">
    <bibliotecario>
      <nombre>Juan</nombre>
      <apellidos>Pérez García</apellidos>
    </bibliotecario>
    <fecha>
      <prestamo>2024-10-26</prestamo>
      <devolucion>2024-11-10</devolucion>
    </fecha>
  </prestamo>
</prestamos>

```

No errors were found

The following files have been uploaded so far:

XML document: 

2) De acuerdo al siguiente documento XML, realizar lo siguiente:

```
<?xml version="1.0"?>
<CatalogoPelículas>
  <Película>
    <Titulo>The Matrix</Titulo>
    <Duración>136</Duración>
    <Genero>Sci-Fi and Fantasy</Genero>
    <Actores>
      <Actor>Keanu Reeves</Actor>
      <Actor>Laurence Fishburne</Actor>
      <Actor>Carrie Ann Moss</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>Wachowski Sisters</Director>
    <Formato>DVD</Formato>
  </Película>
  <Película>
    <Titulo>Titanic</Titulo>
    <Duración>194</Duración>
    <Genero>Drama</Genero>
    <Actores>
      <Actor>Leonardo DiCaprio</Actor>
      <Actor>Kate Winslet</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>James Cameron</Director>
    <Formato>DVD</Formato>
  </Película>
  <Película>
    <Titulo>The Sixth Sense</Titulo>
    <Duración>106</Duración>
    <Genero>Thriller</Genero>
    <Actores>
      <Actor>Bruce Willis</Actor>
      <Actor>Haley Joel Osment</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>M. Night Shyamalan</Director>
    <Formato>VHS</Formato>
  </Película>
</CatalogoPelículas>
```

a. Verificar que sea un documento bien formado.

El documento está bien formado porque cumple con las 10 reglas y pautas de XML con respecto a la sintaxis de la misma. Lo comprobamos con la página www.xmlvalidation.com

No errors were found

The following files have been uploaded so far:

[XML document:](#) 0

b. Crear el documento DTD respectivo:

```
<!ELEMENT CatalogoPeliculas (Pelicula*)>
<!ELEMENT Pelicula (Titulo, Duracion, Genero, Actores, Fecha, Director,
Formato)>
<!ELEMENT Titulo (#PCDATA)>
<!ELEMENT Duracion (#PCDATA)>
<!ELEMENT Genero (#PCDATA)>
<!ELEMENT Actores (Actor+)>
<!ELEMENT Actor (#PCDATA)>
<!ELEMENT Fecha (#PCDATA)>
<!ELEMENT Director (#PCDATA)>
<!ELEMENT Formato (#PCDATA)>
```

c. Validar XML con el DTD creado.

```
<?xml version="1.0"?>
<!DOCTYPE CatalogoPeliculas SYSTEM "CagatoloPelis.dtd">
<CatalogoPeliculas>
  <Pelicula>
    <Titulo>The Matrix</Titulo>
    <Duracion>136</Duracion>
    <Genero>Sci-Fi and Fantasy</Genero>
    <Actores>
      <Actor>Keanu Reeves</Actor>
      <Actor>Laurence Fishburne</Actor>
      <Actor>Carrie Ann Moss</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>Wachowski Sisters</Director>
    <Formato>DVD</Formato>
  </Pelicula>
  <Pelicula>
    <Titulo>Titanic</Titulo>
    <Duracion>194</Duracion>
    <Genero>Drama</Genero>
```

```
<Actores>
  <Actor>Leonardo DiCaprio</Actor>
  <Actor>Kate Winslet</Actor>
</Actores>
<Fecha>1999</Fecha>
<Director>James Cameron</Director>
<Formato>DVD</Formato>
</Pelicula>
<Pelicula>
  <Titulo>The Sixth Sense</Titulo>
  <Duracion>106</Duracion>
  <Genero>Thriller</Genero>
  <Actores>
    <Actor>Bruce Willis</Actor>
    <Actor>Haley Joel Osment</Actor>
  </Actores>
  <Fecha>1999</Fecha>
  <Director>M. Night Shyamalan</Director>
  <Formato>VHS</Formato>
</Pelicula>
</CatalogoPelículas>
```

Please copy your XML document in here:

```
<?xml version="1.0"?>
<!DOCTYPE CatalogoPeliculas SYSTEM "CagatoloPelis.dtd">
<CatalogoPeliculas>
  <Pelicula>
    <Titulo>The Matrix</Titulo>
    <Duracion>136</Duracion>
    <Genero>Sci-Fi and Fantasy</Genero>
    <Actores>
      <Actor>Keanu Reeves</Actor>
      <Actor>Laurence Fishburne</Actor>
      <Actor>Carrie Ann Moss</Actor>
    </Actores>
  </Pelicula>
</CatalogoPeliculas>
```

Or upload it:

No se seleccionó un archivo.

The validation check is performed against any XML schema or DTD declared inside the XML document.

If neither an XML schema nor a DTD is declared, only a syntax check is performed.

To validate the XML document against an external XML schema, click below.

☐ Validate against external XML schema

The file CagatoloPelis.dtd is being referenced. Please copy it in here, so that the validation can continue:

```
<!ELEMENT CatalogoPeliculas (Pelicula*)>
<!ELEMENT Pelicula (Titulo, Duracion, Genero, Actores, Fecha, Director, Formato)>
<!ELEMENT Titulo (#PCDATA)>
<!ELEMENT Duracion (#PCDATA)>
<!ELEMENT Genero (#PCDATA)>
<!ELEMENT Actores (Actor+)>
<!ELEMENT Actor (#PCDATA)>
<!ELEMENT Fecha (#PCDATA)>
<!ELEMENT Director (#PCDATA)>
<!ELEMENT Formato (#PCDATA)>
```

Or upload it:

No se seleccionó un archivo.

The following files have been uploaded so far:

[XML document:](#) 

Click on any file name if you want to edit the file.

No errors were found

The following files have been uploaded so far:

[XML document:](#) 

[CagatoloPelis.dtd](#) 

Click on any file name if you want to edit the file.

3) Escribir un documento XML que pueda ser validado con el siguiente DTD:

```
<!ELEMENT Libro (Titulo, Contenido, Copyright)>
<!ELEMENT Titulo (#PCDATA)>
<!ELEMENT Contenido ((Capitulo+, Separacion?)+)>
<!ELEMENT Capitulo (Tema, Seccion+)>
<!ATTLIST Capitulo materia (XML|Java) "Java">
<!ELEMENT Tema (#PCDATA)>
<!ELEMENT Seccion (#PCDATA)>
<!ATTLIST Seccion apartados CDATA #REQUIRED dificil (si|no) "no">
<!ELEMENT Separacion EMPTY>
<!ELEMENT Copyright (#PCDATA)>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Libro SYSTEM "p3.dtd">
<Libro>
  <Titulo>LibroGenérico</Titulo>
  <Contenido>
    <Capitulo materia="Java">
      <Tema>Tema1</Tema>
      <Seccion apartados="1: Apartado1"
dificil="no">Sección1</Seccion>
    </Capitulo>
    <Separacion></Separacion>
    <Capitulo>
      <Tema>Tema2</Tema>
      <Seccion apartados="2: Apartado2">Sección2</Seccion>
    </Capitulo>
    <Capitulo materia="XML">
      <Tema>Tema3</Tema>
      <Seccion apartados="3: Apartado3"
dificil="si">Sección3</Seccion>
    </Capitulo>
```



```
<Capitulo materia="XML">
  <Tema>Tema4</Tema>
  <Seccion apartados="4: Apartado4"
dificil="si">Sección4</Seccion>
</Capitulo>
<Separacion></Separacion>
<Capitulo>
  <Tema>Tema5</Tema>
  <Seccion apartados="5: Apartado5">Sección5</Seccion>
  <Seccion apartados="6: Apartado6"
dificil="si">Sección6</Seccion>
</Capitulo>
</Contenido>
<Copyright>Todos los derechos reservados a la Academia de Costa del
Pacífico 2024®</Copyright>
</Libro>
```

Please copy your XML document in here:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Libro SYSTEM "p3.dtd">
<Libro>
  <Titulo>LibroGenérico</Titulo>
  <Contenido>
    <Capitulo materia="Java">
      <Tema>Tema1</Tema>
      <Seccion apartados="1: Apartado1" dificil="no">Sección1</Seccion>
    </Capitulo>
    <Separacion></Separacion>
```

The file p3.dtd is being referenced. Please copy it in here, so that the validation can continue:

```
<!ELEMENT Libro (Titulo, Contenido, Copyright)>
<!ELEMENT Titulo (#PCDATA)>
<!ELEMENT Contenido ((Capitulo+, Separacion?)+)>
<!ELEMENT Capitulo (Tema, Seccion+)>
<!ATTLIST Capitulo materia (XML|Java) "Java">
<!ELEMENT Tema (#PCDATA)>
<!ELEMENT Seccion (#PCDATA)>
<!ATTLIST Seccion apartados CDATA #REQUIRED difícil (si|no) "no">
<!ELEMENT Separacion EMPTY>
<!ELEMENT Copyright (#PCDATA)>
```

Or upload it:

Seleccionar archivo Ningún archivo seleccionado

continue validation

No errors were found

The following files have been uploaded so far:

[XML document](#)

[p3.dtd](#)

Click on any file name if you want to edit the file.

4) En la tienda de alquiler de películas se empezaron a recibir catálogos de manera más dinámica y de distintos proveedores, todos como documentos XML, lo que implica realizar algunas modificaciones a la base de datos. En la base de datos "sakila" realizar las siguientes acciones:

- a. Crear una nueva tabla con el siguiente esquema:
 - **nuevos_catalogos**(id: entero, fecha_alta: fecha, catalogo: texto)

```
mysql> CREATE TABLE nuevos_catalogos (id INT, fecha_alta DATE, catalogo TEXT);
Query OK, 0 rows affected (0,06 sec)
```

```
mysql>
```

- b. Agregar un registro a la tabla creada de manera de dar de alta el siguiente catálogo:

```
<CatalogoPeliculas>
  <Pelicula>
```

```

<Titulo>The Matrix</Titulo>
<Duracion>136</Duracion>
<Genero>Sci-Fi and Fantasy</Genero>
<Actores>
  <Actor>Keanu Reeves</Actor>
  <Actor>Laurence Fishburne</Actor>
  <Actor>Carrie Ann Moss</Actor>
</Actores>
<Fecha>1999</Fecha>
<Director>Wachowski Brothers</Director>
<Formato>DVD</Formato>
</Pelicula>
<Pelicula>
  <Titulo>Titanic</Titulo>
  <Duracion>194</Duracion>
  <Genero>Drama</Genero>
  <Actores>
    <Actor>Leonardo DiCaprio</Actor>
    <Actor>Kate Winslet</Actor>
  </Actores>
  <Fecha>1999</Fecha>
  <Director>James Cameron</Director>
  <Formato>DVD</Formato>
</Pelicula>
<Pelicula>
  <Titulo>The Sixth Sense</Titulo>
  <Duracion>106</Duracion>
  <Genero>Thriller</Genero>
  <Actores>
    <Actor>Bruce Willis</Actor>
    <Actor>Haley Joel Osment</Actor>
  </Actores>
  <Fecha>1999</Fecha>
  <Director>M. Night Shyamalan</Director>
  <Formato>VHS</Formato>
</Pelicula>
</CatalogoPelículas>

```

Para cargar el catálogo, optamos por tener un archivo con extensión .xml en la máquina virtual de Multipass para luego cargarlo al base de datos con la opción "LOAD_FILE". Lo veremos a continuación...

```
ubuntu@mv-sgbd ~/máquina local/Scripts SQL TP 5$ sudo vim /var/lib/mysql-files/p4.xml
```

```
1/2 + [ ] [ ] Tilix: sudo vim /var/lib/mysql-files/p4.xml

<CatalogoPelículas>
  <Película>
    <Titulo>The Matrix</Titulo>
    <Duracion>136</Duracion>
    <Genero>Sci-Fi and Fantasy</Genero>
    <Actores>
      <Actor>Keanu Reeves</Actor>
      <Actor>Laurence Fishburne</Actor>
      <Actor>Carrie Ann Moss</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>Wachowski Brothers</Director>
    <Formato>DVD</Formato>
  </Película>
  <Película>
    <Titulo>Titanic</Titulo>
    <Duracion>194</Duracion>
    <Genero>Drama</Genero>
    <Actores>
      <Actor>Leonardo DiCaprio</Actor>
      <Actor>Kate Winslet</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>James Cameron</Director>
    <Formato>DVD</Formato>
  </Película>
  <Película>
    <Titulo>The Sixth Sense</Titulo>
    <Duracion>106</Duracion>
    <Genero>Thriller</Genero>
    <Actores>
      <Actor>Bruce Willis</Actor>
      <Actor>Haley Joel Osment</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>M. Night Shyamalan</Director>
    <Formato>VHS</Formato>
  </Película>
</CatalogoPelículas>
```

Vemos que el archivo xml está bajo el directorio /var/lib/mysql-files con el nombre "p4.xml"

```
mysql> SHOW VARIABLES LIKE 'secure_file_priv';
+-----+-----+
| Variable_name | Value                               |
+-----+-----+
| secure_file_priv | /var/lib/mysql-files/             |
+-----+-----+
1 row in set (0,01 sec)

mysql>
```

Verificamos que esté configurado el directorio correcto para "secure_file_priv", variable la cual tiene que estar activada para poder cargar luego los datos con la opción "LOAD_FILE".

```
mysql> INSERT INTO nuevos_catalogos (id, fecha_alta, catalogo) VALUES (1, DATE(NOW()), LOAD_FILE('/var/lib/mysql-files/p4.xml'));
Query OK, 1 row affected (0,04 sec)

mysql> █
```

Hacemos la inserción del catálogo.

```
mysql> SELECT * FROM nuevos_catalogos\G
***** 1. ROW *****
      id: 1
fecha_alta: 2024-10-26
catalogo: <CatalogoPelículas>
  <Película>
    <Titulo>The Matrix</Titulo>
    <Duracion>136</Duracion>
    <Genero>Sci-Fi and Fantasy</Genero>
    <Actores>
      <Actor>Keanu Reeves</Actor>
      <Actor>Laurence Fishburne</Actor>
      <Actor>Carrie Ann Moss</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>Wachowski Brothers</Director>
    <Formato>DVD</Formato>
  </Película>
  <Película>
    <Titulo>Titanic</Titulo>
    <Duracion>194</Duracion>
    <Genero>Drama</Genero>
    <Actores>
      <Actor>Leonardo DiCaprio</Actor>
      <Actor>Kate Winslet</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>James Cameron</Director>
    <Formato>DVD</Formato>
  </Película>
  <Película>
    <Titulo>The Sixth Sense</Titulo>
    <Duracion>106</Duracion>
    <Genero>Thriller</Genero>
    <Actores>
      <Actor>Bruce Willis</Actor>
      <Actor>Haley Joel Osment</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>M. Night Shyamalan</Director>
    <Formato>VHS</Formato>
  </Película>
</CatalogoPelículas>

1 row in set (0,00 sec)

mysql> █
```

Comprobamos que se haya cargado correctamente.

- c. Utilizar las funciones XML disponibles en MySQL para resolver las siguientes consignas...

Aclaración! Para completar cada una de las consignas, utilizaremos el siguiente script:

```
Sistemas de Gestión de Bases de Datos/Compartido con Multipass/Scripts SQL TP 5 git:main*  
> nvim p4.sql
```

```
1  
2 INSERT INTO nuevos_catalogos (id, fecha_alta, catalogo) VALUES (1, DATE(NOW()), LOAD_FILE('/var/  
lib/mysql-files/p4.xml'));  
3  
4 -- c) Utilizar las funciones XML disponibles en MySQL para:  
5 -- i. Obtener fecha de alta y actores de la película Titanic.  
6  
7 -- Versión 1:-  
8 SELECT fecha_alta, EXTRACTVALUE(catalogo, '/CatalogoPeliculas/Pelicula[Titulo="Titanic"]/Actores/  
Actor') AS 'Actores de la película "El Titanic"'  
9 FROM nuevos_catalogos;  
10  
11 -- Versión 2:-  
12 SELECT fecha_alta, ExtractValue(catalogo, '//Pelicula[Titulo="Titanic"] /Actores/Actor/text()') AS  
'Actores de la película "El Titanic"' FROM nuevos_catalogos;  
13  
14 -- ii. Listar título de las películas en las que actúa Leonardo DiCaprio.  
15  
16 SELECT EXTRACTVALUE(catalogo, '/CatalogoPeliculas/Pelicula[Actores/Actor="Leonardo DiCaprio"]/  
Titulo') AS 'Películas de Leonardo DiCaprio'  
17 FROM nuevos_catalogos;  
18  
19 -- iii. Obtener la duración de las películas de género Thriller.  
20  
21 SELECT EXTRACTVALUE(catalogo, '/CatalogoPeliculas/Pelicula[Genero="Thriller"]/Duracion') AS  
'Duración de las películas Thriller'  
22 FROM nuevos_catalogos;  
23  
24 -- iv. Actualizar las películas en formato "VHS" a "Blue-Ray".  
25  
26 SELECT UPDATEXML(catalogo, '/CatalogoPeliculas/Pelicula[Formato="VHS"]/Formato', '<Formato>Blue-  
Ray</Formato>') AS 'Películas en Blue-Ray\n'  
27 FROM nuevos_catalogos\G
```

```
Sistemas de Gestión de Bases de Datos/Compartido con Multipass/Scripts SQL TP 5 git:main*  
> multipass exec -n mv-sgbd -- mysql -u George\ Harrison -pmust_pass sakila -e "source /home/ubuntu/máquin  
a local/Scripts SQL TP 5/p4.sql"
```

Ejecutamos el script en la base de datos de la máquina virtual.

1. Obtener fecha de alta y actores de la película Titanic.

```
Sistemas de Gestión de Bases de Datos/Compartido con Multipass/Scripts SQL TP 5 git:main*  
> multipass exec -n mv-sgbd -- mysql -u George\ Harrison -pmust_pass sakila -e "source /home/ubuntu/máquin  
a local/Scripts SQL TP 5/p4.sql"  
mysql: [Warning] Using a password on the command line interface can be insecure.  
+-----+-----+  
| fecha_alta | Actores de la película "El Titanic" |  
+-----+-----+  
| 2024-10-28 | Leonardo DiCaprio Kate Winslet |  
+-----+-----+  
+-----+-----+  
| fecha_alta | Actores de la película "El Titanic" |  
+-----+-----+  
| 2024-10-28 | Leonardo DiCaprio Kate Winslet |  
+-----+-----+
```

2. Listar título de las películas en las que actúa Leonardo DiCaprio.

```
+-----+
| Películas de Leonardo DiCaprio |
+-----+
| Titanic                         |
+-----+
```

3. Obtener la duración de las películas de género Thriller.

```
+-----+
| Duración de las películas Thriller |
+-----+
| 106                                |
+-----+
```

4. Actualizar las películas en formato "VHS" a "Blue-Ray".

```
***** 1. row *****
Películas en Blue-Ray
: <CatalogoPelículas>
  <Película>
    <Titulo>The Matrix</Titulo>
    <Duracion>136</Duracion>
    <Genero>Sci-Fi and Fantasy</Genero>
    <Actores>
      <Actor>Keanu Reeves</Actor>
      <Actor>Laurence Fishburne</Actor>
      <Actor>Carrie Ann Moss</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>Wachowski Brothers</Director>
    <Formato>DVD</Formato>
  </Película>
  <Película>
    <Titulo>Titanic</Titulo>
    <Duracion>194</Duracion>
    <Genero>Drama</Genero>
    <Actores>
      <Actor>Leonardo DiCaprio</Actor>
      <Actor>Kate Winslet</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>James Cameron</Director>
    <Formato>DVD</Formato>
  </Película>
  <Película>
    <Titulo>The Sixth Sense</Titulo>
    <Duracion>106</Duracion>
    <Genero>Thriller</Genero>
    <Actores>
      <Actor>Bruce Willis</Actor>
      <Actor>Haley Joel Osment</Actor>
    </Actores>
    <Fecha>1999</Fecha>
    <Director>M. Night Shyamalan</Director>
    <Formato>Blue-Ray</Formato>
  </Película>
</CatalogoPelículas>
```


5) Instale la base de datos de ejemplo “world_x” y realice las siguientes acciones:

```
~/OneDrive/Facultad/3º año/Sistemas de Gestión de Bases de Datos/Compartido con Multipass
> wget http://downloads.mysql.com/docs/world_x-db.zip
URL transformed to HTTPS due to an HSTS policy
--2024-10-26 21:05:45-- https://downloads.mysql.com/docs/world_x-db.zip
Resolviendo downloads.mysql.com (downloads.mysql.com)... 2600:1419:1200:881::2e31, 2600:1419:1200:883::2e31, 104.10
4.34.126
Conectando con downloads.mysql.com (downloads.mysql.com)[2600:1419:1200:881::2e31]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 99539 (97K) [application/octet-stream]
Grabando a: «world_x-db.zip»

world_x-db.zip          100%[=====] 97,21K  204KB/s  en 0,5s

2024-10-26 21:05:47 (204 KB/s) - «world_x-db.zip» guardado [99539/99539]

~/OneDrive/Facultad/3º año/Sistemas de Gestión de Bases de Datos/Compartido con Multipass
> unzip world_x-db.zip
Archive:  world_x-db.zip
  creating: world_x-db/
  inflating: world_x-db/README.txt
  inflating: world_x-db/world_x.sql

~/OneDrive/Facultad/3º año/Sistemas de Gestión de Bases de Datos/Compartido con Multipass
> [Icono] Terminar Sesión
```

Descargamos y descomprimos los archivos para crear la base de datos “world_x”.

```
ubuntu@mv-sgbd ~/máquina local/world_x-db$ mysql -u George\ Harrison -pmust_pass < world_x.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
ubuntu@mv-sgbd ~/máquina local/world_x-db$

ubuntu@mv-sgbd ~/máquina local/Scripts SQL TP 5$ mysql -u George\ Harrison -pmust_pass -e "USE world_x; SHOW DATA
BASES; SHOW TABLES"
mysql: [Warning] Using a password on the command line interface can be insecure.
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sakila |
| sys |
| world_x |
+-----+
+-----+
| Tables_in_world_x | la base de datos "world_x"
+-----+
| city |
| country |
| countryinfo |
| countrylanguage |
+-----+
```

Comprobamos que la base de datos haya sido creada correctamente.

a. Listar los nombres de las ciudades y su cantidad de habitantes.

```
mysql> SELECT Name AS Ciudad, JSON_EXTRACT(Info, '$.Population') AS Poblacion FROM city;
```

Ciudad	Poblacion
Kabul	1780000
Qandahar	237500
Herat	186800
Mazar-e-Sharif	127800
Amsterdam	731200
Rotterdam	593321
Haag	440900
Utrecht	234323
Eindhoven	201843
Tilburg	193238
Groningen	172701
Breda	160398
Apeldoorn	153491
Nijmegen	152463
Enschede	149544
Haarlem	148772
Almere	142465
Arnhem	138020
Zaanstad	135621
's-Hertogenbosch	129170
Amersfoort	126270
Maastricht	122087
Dordrecht	119811
Leiden	117196
Haarlemmermeer	110722
Zoetermeer	110214
Emmen	105853
Zwolle	105819
Ede	101574
Delft	95268

```
4079 rows in set (0.01 sec)
```

b. Encuentre las ciudades con una población superior a 1.000.000 de habitantes.

```
mysql> SELECT Name AS Ciudad, JSON_EXTRACT(Info, '$.Population') AS Poblacion
-> FROM city
-> WHERE JSON_EXTRACT(Info, '$.Population') > 1000000;
```

Ciudad	Poblacion
Kabul	1780000
Alger	2168000
Luanda	2022000
Buenos Aires	2982146
La Matanza	1266461
Córdoba	1157507
Yerevan	1248700
Sydney	3276207
Melbourne	2865329
Brisbane	1291117
Perth	1096829
Baku	1787800
Dhaka	3612850
Chittagong	1392860
São Paulo	9968485
Rio de Janeiro	5598953
Salvador	2302832
Belo Horizonte	2139125
Fortaleza	2097757

Harare	1410000
--------	---------

237 rows in set (0.01 sec)

c. Mostrar un listado “entendible” de la información registrada de los 10 últimos países ordenados por código (campo Code).

```
mysql> SELECT
-> c.Capital AS ID_Capital,
-> JSON_EXTRACT(ci.doc, '$.Name') AS Nombre_País,
-> JSON_EXTRACT(ci.doc, '$.geography.Region') AS Región,
-> JSON_EXTRACT(ci.doc, '$.geography.Continent') AS Continente,
-> JSON_EXTRACT(ci.doc, '$.government.HeadOfState') AS Jefe_de_Estado,
-> JSON_EXTRACT(ci.doc, '$.government.GovernmentForm') AS Forma_de_Gobierno,
-> JSON_EXTRACT(ci.doc, '$.demographics.Population') AS Población
-> FROM country AS c INNER JOIN countryinfo AS ci ON c.Code = JSON_EXTRACT(ci.doc, '$.Code')
-> ORDER BY c.Code DESC
-> LIMIT 10;
```

ID_Capital	Nombre_País	Región	Continente	Jefe_de_Estado	Forma_de_Gobierno	Población
4068	"Zimbabwe"	"Eastern Africa"	"Africa"	"Robert G. Mugabe"	"Republic"	11669000
3162	"Zambia"	"Eastern Africa"	"Africa"	"Frederick Chiluba"	"Republic"	9169000
716	"South Africa"	"Southern Africa"	"Africa"	"Thabo Mbeki"	"Republic"	40377000
1792	"Yugoslavia"	"Southern Europe"	"Europe"	"Vojislav Koštunica"	"Federal Republic"	10640000
1780	"Yemen"	"Middle East"	"Asia"	"Ali Abdallah Salih"	"Republic"	18112000
3169	"Samoa"	"Polynesia"	"Oceania"	"Malietoa Tanumafili II"	"Parliamentary Monarchy"	130000
3536	"Wallis and Futuna"	"Polynesia"	"Oceania"	"Jacques Chirac"	"Nonmetropolitan Territory of France"	15000
3537	"Vanuatu"	"Melanesia"	"Oceania"	"John Bani"	"Republic"	190000
3770	"Vietnam"	"Southeast Asia"	"Asia"	"Trần Duc Luong"	"Socialistic Republic"	79832000
4067	"Virgin Islands, U.S."	"Caribbean"	"North America"	"George W. Bush"	"US Territory"	93000

10 rows in set (0.00 sec)

d. Obtener las claves del documento json (campo doc) que se registran en countryinfo.

```
mysql> SELECT
-> JSON_KEYS(doc) AS Claves_Nivel_Superior,
-> JSON_KEYS(JSON_EXTRACT(doc, '$.geography')) AS Claves_geography,
-> JSON_KEYS(JSON_EXTRACT(doc, '$.government')) AS Claves_government,
-> JSON_KEYS(JSON_EXTRACT(doc, '$.demographics')) AS Claves_demographics
-> FROM countryinfo LIMIT 1;
```

Claves_Nivel_Superior	Claves_geography	Claves_government	Claves_demographics
["GNP", "_id", "Code", "Name", "IndepYear", "geography", "government", "demographics"]	["Region", "Continent", "SurfaceArea"]	["HeadOfState", "GovernmentForm"]	["Population", "LifeExpectancy"]

1 row in set (0.00 sec)

e. Analizar la estructura y contenidos del campo doc y agregar a la tabla countryinfo una restricción CHECK que permita validar los documentos de acuerdo a un esquema json. Comprobar el funcionamiento de la nueva restricción.

```
mysql> ALTER TABLE countryinfo
-> ADD CONSTRAINT chk_valid_json CHECK (
->     JSON_UNQUOTE(JSON_EXTRACT(doc, '$.Code')) IS NOT NULL
->     AND CHAR_LENGTH(JSON_UNQUOTE(JSON_EXTRACT(doc, '$.Code'))) = 3
->     AND JSON_UNQUOTE(JSON_EXTRACT(doc, '$.Name')) IS NOT NULL
-> );
Query OK, 240 rows affected (0.06 sec)
Records: 240  Duplicates: 0  Warnings: 0
```

Para un registro con esquema JSON inválido:

```
mysql> INSERT INTO countryinfo (doc) VALUES (
->     '{
->         "GNP": null,
->         "_id": null,
->         "Code": null,
->         "Name": null,
->         "IndepYear": null,
->         "geography": {
->             "Region": null,
->             "Continent": null,
->             "SurfaceArea": null
->         },
->         "government": {
->             "HeadOfState": null,
->             "GovernmentForm": null
->         },
->         "demographics": {
->             "Population": null,
->             "LifeExpectancy": null
->         }
->     }'
-> );
ERROR 3819 (HY000): Check constraint 'chk_valid_json' is violated.
```

Para un registro con esquema JSON válido:

```
mysql> INSERT INTO countryinfo (doc) VALUES (  
-> '{  
->   "GNP": null,  
->   "_id": null,  
->   "Code": "WWW",  
->   "Name": "País de prueba",  
->   "IndepYear": null,  
->   "geography": {  
->     "Region": null,  
->     "Continent": null,  
->     "SurfaceArea": null  
->   },  
->   "government": {  
->     "HeadOfState": null,  
->     "GovernmentForm": null  
->   },  
->   "demographics": {  
->     "Population": null,  
->     "LifeExpectancy": null  
->   }  
-> }'  
-> );  
Query OK, 1 row affected (0.01 sec)
```

f. Listar los países que no tienen registrado año de declaración de la independencia.

```
mysql> SELECT JSON_PRETTY(doc) AS Info_País  
-> FROM countryinfo  
-> WHERE JSON_EXTRACT(doc, '$.IndepYear') LIKE '%null%'\G  
***** 1. row *****  
Info_País: {  
  "GNP": 828,  
  "_id": "00005de917d80000000000000000",  
  "Code": "ABW",  
  "Name": "Aruba",  
  "IndepYear": null,  
  "geography": {  
    "Region": "Caribbean",  
    "Continent": "North America",  
    "SurfaceArea": 193  
  },  
  "government": {  
    "HeadOfState": "Beatrix",  
    "GovernmentForm": "Nonmetropolitan Territory of The Netherlands"  
  },  
  "demographics": {  
    "Population": 103000,  
    "LifeExpectancy": 78.4000015258789  
  }  
}
```

```

***** 48. row *****
Info_País: {
  "GNP": null,
  "_id": null,
  "Code": "WWW",
  "Name": "País de prueba",
  "IndepYear": null,
  "geography": {
    "Region": null,
    "Continent": null,
    "SurfaceArea": null
  },
  "government": {
    "HeadOfState": null,
    "GovernmentForm": null
  },
  "demographics": {
    "Population": null,
    "LifeExpectancy": null
  }
}
48 rows in set (0.00 sec)

```

g. Listar los datos demográficos y la población de su capital para los 10 países con menor superficie.

```

mysql> SELECT city.Name,
  -> JSON_UNQUOTE(JSON_EXTRACT(countryinfo.doc, '$.Name')) AS Nombre,
  -> JSON_UNQUOTE(JSON_EXTRACT(countryinfo.doc, '$.demographics.Population')) AS poblacion,
  -> JSON_UNQUOTE(JSON_EXTRACT(countryinfo.doc, '$.demographics.LifeExpectancy')) AS esperanza_vida,
  -> JSON_UNQUOTE(JSON_EXTRACT(city.Info, '$.Population')) AS poblacion_capital,
  -> JSON_UNQUOTE(JSON_EXTRACT(countryinfo.doc, '$.geography.SurfaceArea')) AS area
  -> FROM countryinfo INNER JOIN country ON country.Name = JSON_UNQUOTE(JSON_EXTRACT(countryinfo.doc, '$.Name'))
  -> INNER JOIN city ON country.Code = city.CountryCode AND country.Capital = city.ID
  -> ORDER BY CAST(JSON_UNQUOTE(JSON_EXTRACT(countryinfo.doc, '$.geography.SurfaceArea')) AS UNSIGNED) ASC LIMIT 10\G
***** 1. row *****
      Name: Città del Vaticano
      Nombre: Holy See (Vatican City State)
      poblacion: 1000
      esperanza_vida: null
      poblacion_capital: 455
      area: 0.4000000059604645
***** 2. row *****
      Name: Monaco-Ville
      Nombre: Monaco
      poblacion: 34000
      esperanza_vida: 78.80000305175781
      poblacion_capital: 1234
      area: 1.5

```

```

***** 9. row *****
      Name: Kingston
      Nombre: Norfolk Island
      poblacion: 2000
      esperanza_vida: null
      poblacion_capital: 800
      area: 36
***** 10. row *****
      Name: Adamstown
      Nombre: Pitcairn
      poblacion: 50
      esperanza_vida: null
      poblacion_capital: 42
      area: 49
10 rows in set, 2 warnings (0.00 sec)

```

h. Agregar una clave al campo Info de la tabla city para representar el código de área telefónico 362 de la ciudad de Resistencia.

```
mysql> UPDATE city
  -> SET Info = JSON_INSERT(Info, '$.Codigo_area', 362)
  -> WHERE Name LIKE 'Resistencia';
Query OK, 0 rows affected (0.01 sec)
Rows matched: 1  Changed: 0  Warnings: 0
```

```
mysql> SELECT JSON_PRETTY(Info)
  -> FROM city
  -> WHERE city.Name LIKE '%Resistencia%';
+-----+
| JSON_PRETTY(Info) |
+-----+
| {
  "Population": 229212,
  "Codigo_area": 362
} |
+-----+
1 row in set (0.00 sec)
```

i. Listar la información de los países cuya forma de gobierno es república federal ("GovernmentForm": "Federal Republic").

```
mysql> SELECT JSON_PRETTY(doc) AS 'Información del País'
  -> FROM countryinfo
  -> WHERE JSON_EXTRACT(doc, '$.government.GovernmentForm') LIKE '%Federal%Republic%\G
***** 1. row *****
Información del País: {
  "GNP": 340238,
  "_id": "00005de917d800000000000000000008",
  "Code": "ARG",
  "Name": "Argentina",
  "IndepYear": 1816,
  "geography": {
    "Region": "South America",
    "Continent": "South America",
    "SurfaceArea": 2780400
  },
  "government": {
    "HeadOfState": "Fernando de la Rúa",
    "GovernmentForm": "Federal Republic"
  },
  "demographics": {
    "Population": 46044703,
    "LifeExpectancy": 75.0999984741211
  }
}
```

```

***** 15. row *****
Información del País: {
  "GNP": 17000,
  "_id": "00005de917d800000000000000eb",
  "Code": "YUG",
  "Name": "Yugoslavia",
  "IndepYear": 1918,
  "geography": {
    "Region": "Southern Europe",
    "Continent": "Europe",
    "SurfaceArea": 102173
  },
  "government": {
    "HeadOfState": "Vojislav Koštunica",
    "GovernmentForm": "Federal Republic"
  },
  "demographics": {
    "Population": 10640000,
    "LifeExpectancy": 72.4000015258789
  }
}
15 rows in set (0.01 sec)

```

j. Actualizar la población de Argentina a 46.044.703 habitantes.

```

mysql> UPDATE countryinfo
-> SET doc = JSON_SET(doc, '$.demographics.Population', 46044703)
-> WHERE JSON_EXTRACT(doc, '$.Code') LIKE '%ARG%';
Query OK, 0 rows affected (0.00 sec)
Rows matched: 1 Changed: 0 Warnings: 0

```

```

mysql> SELECT JSON_PRETTY(doc)
-> FROM countryinfo
-> WHERE JSON_EXTRACT(doc, '$.Name') LIKE '%Argentina%';
+-----+
+
+ | JSON_PRETTY(doc)
+ |
+-----+
+
+ | {
+ |   "GNP": 340238,
+ |   "_id": "00005de917d80000000000000008",
+ |   "Code": "ARG",
+ |   "Name": "Argentina",
+ |   "IndepYear": 1816,
+ |   "geography": {
+ |     "Region": "South America",
+ |     "Continent": "South America",
+ |     "SurfaceArea": 2780400
+ |   },
+ |   "government": {
+ |     "HeadOfState": "Fernando de la Rúa",
+ |     "GovernmentForm": "Federal Republic"
+ |   },
+ |   "demographics": {
+ |     "Population": 46044703,
+ |     "LifeExpectancy": 75.0999984741211
+ |   }
+ | }
+-----+
+
+ 1 row in set (0.00 sec)

```